

APPLICANT(S): AIZIKOWITZ, Jacob et al.
SERIAL NO.: 09/775,501
FILED: February 5, 2001
Page 2

AMENDMENTS TO THE CLAIMS

Please add or amend the claims to read as follows, and cancel without prejudice or disclaimer to resubmission in a divisional or continuation application claims indicated as cancelled:

1. (Original) A production system for producing a VIPDL output stream from a dynamic document and a recipients list comprising:
 - a data iterator for iterating through said recipients list one record at a time and computing values for each dynamic object in said dynamic document for said record;
 - a document instantiator for creating an instance document for said record;
 - an output generator operating in tandem with said document instantiator for generating code specifying the rendering of said document instance; and
 - a merge component for generating said VIPDL output stream from output of said output generator.
2. (Original) A production system according to claim 1, wherein said VIPDL output stream is defined in an object-oriented (rendering) specification language.
3. (Original) A production system according to claim 2, wherein said object-oriented specification language is VPS.
4. (Original) A production system according to claim 2, wherein said object-oriented specification language is PPML.
5. (Original) A production system according to claim 1, wherein said VIPDL output stream is defined in HTML.

APPLICANT(S): AIZIKOWITZ, Jacob et al.
SERIAL NO.: 09/775,501
FILED: February 5, 2001
Page 3

6. (Original) A production system according to claim 1, and also comprising a content objects buffer for allowing said data iterator and said document instantiator to operate in a producer consumer manner.
7. (Original) A production system according to claim 1, wherein said merge component comprises:
 - a definitions dictionary comprising reusable object names and their definitions;
 - a booklets section comprising layout information for pages of dynamic document instances; and
 - a merge processor for generating said VIPDL output stream from contents of said definitions dictionary and said booklets section.
8. (Original) A production system for producing a VIPDL output stream from a dynamic document and a recipients list comprising:
 - a parts generator for breaking said recipients list into a plurality of parts;
 - a plurality of pipelines for parallel processing of said recipients list;
 - a plurality of definitions dictionaries for receiving reusable object definitions from said plurality of pipelines;
 - a plurality of booklets sections for receiving layout information from said plurality of pipelines; and
 - a merge processor for generating said VIPDL output stream from contents of said plurality of definitions dictionary and said plurality of booklets section.
9. (Currently Amended) A method for producing a VIPDL output stream from a dynamic document and a recipients list comprising the steps of:

APPLICANT(S): AIZIKOWITZ, Jacob et al.
SERIAL NO.: 09/775,501
FILED: February 5, 2001
Page 4

selecting a next record from said recipients list;

generating a document instance from said dynamic document for said
selected record from said recipients list;

~~generating entries in a definition dictionary corresponding to reusable objects
in said dynamic document;~~

generating a rendering specifications section for said generated document
instance;

concatenating said rendering specifications section into said VIPDL output
stream; and

repeating said ~~step~~ steps of selecting, said ~~first~~ step of generating a document
instance, ~~said second step of generating,~~ said ~~third~~ step of generating a rendering
specifications section, and said step of concatenating until said recipients list has
been exhausted; and

~~generating an elements definition section from the information in said
definitions dictionary and pre-pending said elements definition section at the head
of said VIPDL output stream.~~

10. (Currently Amended) The method according to claim 9, further comprising the
steps of generating entries in a definition dictionary corresponding to reusable objects
in said dynamic document and generating elements definition section from the
information in said definitions dictionary and pre-pending said elements definition
section at the head of said VIPDL output stream, wherein said step of selecting a next
record from said recipients list, said ~~first~~ step of generating a document instance; said

APPLICANT(S): AIZIKOWITZ, Jacob et al.
SERIAL NO.: 09/775,501
FILED: February 5, 2001
Page 5

second-step of generating entries in a definition dictionary, said ~~third~~ step of generating a rendering specifications section, said step of concatenating, said step of repeating, and said ~~fourth~~ step of generating elements definition section are performed by pipeline processing.

11. (Currently Amended) A method according to claim 9, wherein said ~~first~~ step of generating a document instance [[also]] comprises the steps of:
 - collapsing all the queries for the set of said dynamic content objects into one query; and
 - computing said one query.
12. (Currently Amended) A method according to claim 9, wherein said ~~first~~ step of generating a document instance [[also]] comprises the steps of:
 - caching views generated by queries; and
 - reusing said cached views.
13. (Original) A method according to claim 9, wherein said VIPDL output stream is defined in an object-oriented specification language.
14. (Original) A method according to claim 13, wherein said object-oriented specification language is VPS.
15. (Original) A method according to claim 13, wherein said object-oriented specification language is PPML.
16. (Original) A method according to claim 9, wherein said VIPDL output stream is defined in HTML.

APPLICANT(S): AIZIKOWITZ, Jacob et al.
SERIAL NO.: 09/775,501
FILED: February 5, 2001
Page 6

17. (Currently Amended) A method for producing a VIPDL output stream from a dynamic document and a recipients list comprising the steps of:

receiving a recipient list;

dividing said recipients list into a plurality of distinct parts;

performing parallel processing to merge said dynamic documents
independently ~~[[for]]~~ with each of said parts; ~~and~~

~~merging a plurality of definition dictionaries and a plurality of rendering~~
~~specifications sections created in said step of performing into one VIPDL output~~
~~stream corresponding to said entire recipients list.~~

18. (Currently Amended) The method according to claim 17, further comprising the
step of merging a plurality of definition dictionaries and a plurality of rendering
specifications sections created in said step of performing into one VIPDL output
stream corresponding to said entire recipients list. wherein said step of performing
~~selecting, said first step of generating, said second step of generating, said third step~~
~~of, said step of concatenating, said step of repeating, and said fourth step of~~
~~generating~~ ~~[[are]]~~ is performed by pipeline processing.

19. (Original) A method according to claim 17, wherein said step of performing
further comprises the step of:

using a gossiping mechanism to reduce redundancy in said plurality of
definition dictionaries.

20. (Currently Amended) A method according to claim 17, wherein said step of
performing further comprises the steps of:

APPLICANT(S): AIZIKOWITZ, Jacob et al.
SERIAL NO.: 09/775,501
FILED: February 5, 2001
Page 7

selecting a next record from said recipients list;

generating a document instance from said dynamic document for said selected record from said recipients list;

generating entries in a definition dictionary corresponding to dynamic content objects in said dynamic document;

generating a rendering specifications section for said generated document instance;

concatenating said rendering specifications section into said VIPDL output stream;

repeating said step of selecting, said ~~first~~ step of generating a document instance; said ~~second~~ step of generating entries in a definition dictionary, said ~~third~~ step of generating a rendering specifications section, said step of concatenating until said recipients list has been exhausted; and

generating an elements definition section from the information in said definitions dictionary and pre-pending said elements definition section at the head of said VIPDL output stream.

21. (Currently Amended) A method according to claim 20, wherein said ~~first~~ step of generating a document instance ~~[[also]]~~ comprises the steps of:

collapsing all the queries for the set of said dynamic content objects into one query; and

computing said one query minimizing joins.

APPLICANT(S): AIZIKOWITZ, Jacob et al.
SERIAL NO.: 09/775,501
FILED: February 5, 2001
Page 8

22. (Currently Amended) A method according to claim 20, wherein said ~~first~~ step of generating a document instance also comprises the steps of:
- collapsing all the queries for the set of said dynamic content objects into one query; and
 - computing said one query minimizing re-selections.
23. (Currently Amended) A method according to claim 20, wherein said ~~first~~ step of generating a document instance also comprises the steps of:
- caching views generated by queries; and
 - reusing said cached views.
24. (Original) A method according to claim 17, wherein said VIPDL output stream is defined in an object-oriented specification language.
25. (Original) A method according to claim 24, wherein said object-oriented specification language is VPS.
26. (Original) A method according to claim 24, wherein said object-oriented specification language is PPML.
27. (Original) A method according to claim 17, wherein said VIPDL output stream is defined in HTML.
- 28-31 (Cancelled)